ProLinx

# HART Driver Manual

August 17, 2004

ProSoft TECHNOLOGY

## Please Read This Notice

The use and configuration of this software requires a reasonable working knowledge of the involved protocols and the application in which they are to be used. For this reason, it is important that those responsible for implementation satisfy themselves that the combination will meet the needs of the application without exposing personnel or equipment to unsafe or inappropriate working conditions.

This manual is provided to assist the user. Every attempt has been made to assure that the information provided is accurate and a true reflection of the product's functionality. In order to assure a complete understanding of the operation of the product, the user should read all applicable ProSoft documentation on the operation of the module and protocol driver.

Under no conditions will ProSoft Technology, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of the product.

Reproduction of the contents of this manual, in whole or in part, without written permission from ProSoft Technology, Inc. is prohibited.

Information in this manual is subject to change without notice and does not represent a commitment on the part of ProSoft Technology, Inc. Improvements and/or changes in this manual or the product may be made at any time. These changes will be made periodically to correct technical inaccuracies or typographical errors.

ProSoft Technology, Inc.
1675 Chester Avenue, 2nd Floor
Bakersfield, CA 93301
(661) 716-5100
(661) 716-5101 Fax
prosoft@prosoft-technology.com

© ProSoft Technology, Inc. 2000, 2001, 2002, 2003, 2004
ProLinx HART Driver Manual
August 17, 2003

# HART Master
# Protocol Driver Manual

## Table of Contents


**1 Functional Overview ..... 5**
1.1 HART Channels ..... 5
1.2 Module Internal Database ..... 6
1.2.1 HART Channel Driver Access to Database ..... 7
1.2.2 Auto-Polling ..... 7
1.2.2.1 Multidrop Mode ..... 11
1.2.3 HART Command List ..... 14

**2 Protocol Functional Specifications ..... 15**
2.1 HART Channel Specifications ..... 15

**3 HART Protocol Specific Configuration File ..... 17**

**4 CFG File: [HART PORT x] Section ..... 18**

**5 CFG File: [HART PORT x COMMANDS] Section ..... 19**
5.1 Command List Overview ..... 19
5.2 Commands Supported by the Module ..... 20
5.3 HART Command Entry Formats ..... 21

**6 Error/Status Data ..... 24**
6.1 Viewing Error and Status Data ..... 24
6.2 Configuration Error Word Table ..... 24
6.3 Port 0 General Status Data ..... 25
6.4 Port 1 General Status Data ..... 26
6.4.1 General Status Data (Ports 2 through 7) ..... 26

**7 Error/Status Data Structure ..... 27**
7.1 HART Command List Error Data ..... 28
7.1.1 Port 0 Command List Errors ..... 28
7.1.2 Port 1 Command List Errors ..... 28
7.2 Slave List Status ..... 28

**8 Module Error Codes ..... 29**
8.1 Protocol Error Codes ..... 29

**Appendix A: HART Command Support ..... 32**
COMMAND 00: Read Unique Identifier ..... 33
COMMAND 01: Read Primary Variable ..... 34
COMMAND 02: Read Current And Percent Of Range ..... 35
COMMAND 03: Read Dynamic Variables ..... 36

COMMAND 06: Write Polling Address ..... 37
COMMAND 12: Read Message ..... 38
COMMAND 13: Read Tag, Descriptor and Date ..... 39
COMMAND 14: Read PV Sensor Info ..... 40
COMMAND 15: Read Output Information ..... 41
COMMAND 16: Read Final Assembly Number ..... 42
COMMAND 17: Write Message ..... 43
COMMAND 18: Write Tag, Descriptor and Date ..... 44
COMMAND 19: Write Final Assembly Number ..... 45
COMMAND 33: Read Transmitter Variables ..... 46
COMMAND 34: Write Damping Value ..... 47
COMMAND 35: Write Range Values ..... 48
COMMAND 36: Set Upper Range Value ..... 49
COMMAND 37: Set Lower Range Value ..... 50
COMMAND 38: Reset Configuration Changed Flag ..... 51
COMMAND 39: EEPROM Control ..... 52
COMMAND 40: Enter Exit Fixed Current Mode ..... 53
COMMAND 41: Perform Transmitter Self Test ..... 54
COMMAND 42: Perform Master Reset ..... 55
COMMAND 43: Set PV Zero ..... 56
COMMAND 44: Write PV Units ..... 57
COMMAND 45: Trim DAC Zero ..... 58
COMMAND 46: Trim DAC Gain ..... 59
COMMAND 47: Write Transfer Function ..... 60
COMMAND 48: Read Additional Transmitter Status ..... 61
COMMAND 49: Write PV Sensor Serial Number ..... 62
COMMAND 50: Read Dynamic Variable Assignments ..... 63
COMMAND 51: Write Dynamic Variable Assignments ..... 64
COMMAND 52: Set Transmitter Variable Zero ..... 65
COMMAND 53: Write Transmitter Variable Units ..... 66
COMMAND 54: Read Transmitter Variable Information ..... 67
COMMAND 55: Write Transmitter Variable Damping Value ..... 68
COMMAND 56: Write Transmitter Variable Sensor Serial Number ..... 69
COMMAND 57: Read Unit Tag Descriptor Date ..... 70
COMMAND 58: Write Unit Tag Descriptor Date ..... 71
COMMAND 59: Write Number Of Response Preambles ..... 72
COMMAND 108: Write Burst Mode Command Number ..... 73
COMMAND 109: Burst Mode Control ..... 74
COMMAND 110: Read All Dynamic Variables ..... 75

**Appendix B – ProLinx HART Field Connections ..... 77**
Network Power Supply Requirements ..... 77
Connections for HART Devices (Types A, B, C and D) ..... 78
ProLinx HART Connection Diagram for Multiple Loop Powered Devices, Digital Signal Only (A, B, or D types) ..... 78
ProLinx HART Connection Diagram for Multiple Loop Powered Devices, Digital and Analog Signal (A or B Types) ..... 79

ProLinx HART Connection Diagram for Multiple Self-Powered Devices, Digital Signal Only (C or D Types) 80
ProLinx HART Connection Diagram for Single Self-Powered Device, Digital and Analog Signal (C Type only) ..... 81
ProLinx HART Connection Diagram for Multiple Single and Loop-Powered Devices, Digital Signal Only (A, B, C, or D Types) ..... 82

# Contact Information

ProLinx Communication Gateways, Inc.
1675 Chester Avenue, 2nd Floor
Bakersfield, CA 93301
661-716-5100
Fax: 661-716-5101
info@prosoft-technology.com
http://www.prosoft-technology.com

# Related Documents & Reference Materials

Several resources are available to assist with the configuration and support of the ProLinx Communication Gateway modules. The following files are available off the ftp site:

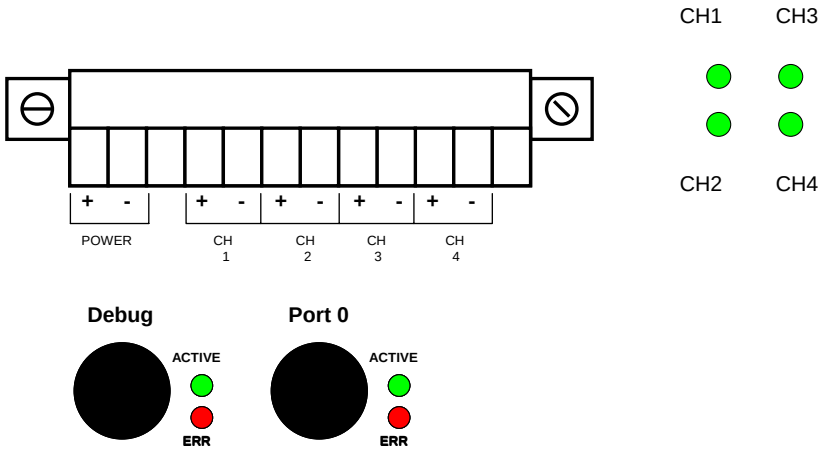
| | | |
|---|---|---|
| Startup Guide | www.prolinxgateways.com/downloads | |
| | Startup_guide_2.20.pdf | ProLinx Communication Gateways, Inc. Startup Guide |

# 1 Functional Overview

The HART Master Protocol exists in 4 and 8 channel implementations. This driver can be configured on an individual channel basis to operate as a HART Master Station and supports all the available HART commands including Universal, Common Practice and Device Specific Commands. Each HART channel is independently configured to interface with the internal database in the module.

## 1.1 HART Channels

The ProLinx module is capable of supporting the HART protocol as a Master on up to 8 channels per interface card. Each of the channels is individually configurable, providing a great deal of flexibility.




The relationship between the port labeling on the front of the ProLinx module and the application is as follows:

| Port Label | Function |
|---|---|
| Debug | Debug/Configuration |
| Port 0 | Communications Port 0 |
| CHANNEL 1 | HART Port 0 |
| CHANNEL 2 | HART Port 1 |
| CHANNEL 3 | HART Port 2 |
| CHANNEL 4 | HART Port 3 |
| CHANNEL 5 | HART Port 4 |
| CHANNEL 6 | HART Port 5 |
| CHANNEL 7 | HART Port 6 |
| CHANNEL 8 | HART Port 7 |

The HART protocol uses the Bell 202 standard frequency shift keying (FSK) signal to communicate at 1200 baud, superimposed at a low level on the 4 to 20 mA analog measurement signal. Having an average value of zero, an FSK signal causes no interference with the analog value. The HART devices are powered from this 4 to 20 mA analog loop.

All HART protocol channels in the Prolinx module generate the Bell 202 FSK signal to communicate in multidrop mode with up to 15 HART devices and provide up to 300 mA supply for the analog loop in multi-drop mode. Refer to Appendix B for ProLinx HART field connection information.

User configured commands determine the HART commands to be issued on each channel to the HART devices. Up to 100 commands can be defined for each port. Data read from the devices are placed in the virtual database. Any write requests or device specific command for the HART slave devices are sourced with data from the virtual database or from a configured constant data block. In the commands it can be specified whether to use the HART device's short or long address. If the long address is selected, the device is polled first with short address to ask for the long one. Then the device is polled with the long address. The module does all this processing of the address automatically.

The module can be configured to place slave devices that are not responding to commands from the master ports at a lower priority. If the module recognizes that a slave device has failed to respond to a message after the user defined retry count, it will mark the slave as "in communication failure" and set the error delay time to the user specified value. Each time that the error delay time expires, the slave will be polled and if the answer is successful, the slave is placed again in an active status. This facility can improve communication throughput on the HART network.

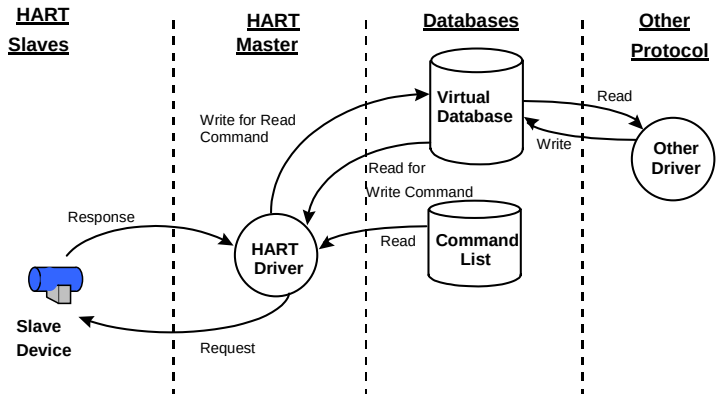
In a HART network, it is possible to have two masters. The Prolinx module fully supports the existence of a second master, but it can reduce the throughput on the HART network. This facility is enabled or disabled in the module's configuration. If the ability to have a second master on the network is disabled, then the maximum throughput is achieved.

## 1.2 Module Internal Database

Central to the functionality of the module is the internal database. This database is shared between all the ports on the module and is used as a conduit to pass information from one device on one network to one or more devices on another network. This permits data from devices on one communication port to be viewed and controlled by devices on another port. In addition to data from the slave and master ports, status and error information generated by the module can also be mapped into the internal database.

## 1.2.1 HART Channel Driver Access to Database

The following diagram details the flow of data between the HART channel drivers and the internal database.




The HART Master driver uses the database in two ways:

1. A read command issued to a slave device by the master driver will return the slave data into the internal database
2. A write command issued to a slave device by the master driver use the data in the internal database to write to the slave device

In addition to data from the master HART channels, detailed status and error information generated by the module can also be mapped into the internal database.

## 1.2.2 Auto-Polling

This feature is enabled by setting the AUTO-POLL CODE (in the configuration file) to a value of P (point-to-point) or M (multidrop). If the value N is entered for the parameter, the auto-poll feature is disabled. When the feature is disabled, the channel will only execute the commands enabled in the user command list. When the auto-polling mode is enabled, the module will automatically acquire data from the HART instruments attached to a channel without the use of user commands. If user commands are present and enabled when the feature is enabled, they will also be executed independent of auto-polling.

With the auto-poll feature enabled, the module automatically generates the following HART commands and stores the data in the module's database at the user-specified location:

| CMD | DESCRIPTION |
|---|---|
| 0 | Read Unique Identifier |
| 3 | Read Current and Four Dynamic Variables |
| 13 | Read Tag, Descriptor and Date |
| 14 | Read PV Sensor Information |
| 15 | Read Output Information |

If the unit is set for point-to-point mode, the module will automatically gather the information for the device with the polling address (short address) of zero and place the data into the database. Each device requires a 50-word database area with the format shown in the following table:

**DEVICE MEMORY LAYOUT**

| DB BYTE OFFSET | TYPE | DESCRIPTION | BYTE CNT | DATA SOURCE | USE OF DATA |
|---|---|---|---|---|---|
| 0 | byte | Auto-polling command status bits | 1 | App | Status |
| 1 | byte | Last first status byte received from device | 1 | Resp | Status |
| 2 | byte | Last second status byte received from device | 1 | Resp | Status |
| 3 | byte | Manufacture ID Code | 1 | CMD 0 | Long Address |
| 4 | byte | Device Type Code | 1 | CMD 0 | Long Address |
| 5 | byte | Minimum number of preambles | 1 | CMD 0 | Msg construction |
| 6 | byte | Universal Command Major Rev # | 1 | CMD 0 | Msg choice |
| 7 | byte | Device Revision Level | 1 | CMD 0 | Info |
| 8 | byte | Software Revision Level | 1 | CMD 0 | Info |
| 9 | byte | Hardware Revision Level/Physical Signaling Code | 1 | CMD 0 | Info |
| 10 | byte | Device Flags | 1 | CMD 0 | Info |
| 11-13 | byte | Device ID | 3 | CMD 0 | Long Address |
| 14 | byte | Minimum number of preambles to be sent with the response message from the slave to the master. | 1 | CMD 0 | |
| 15 | byte | Maximum number of device variables | 1 | CMD 0 | Info |
| 16-17 | word | Configuration Change Counter | 2 | CMD 0 | Info |
| 18 | byte | Extended Field Device Status | 1 | CMD 0 | Info |
| 19 | byte | Primary variable units code | 1 | CMD 3 | Cfg |
| 20 | byte | Secondary variable units code | 1 | CMD 3 | Cfg |
| 21 | byte | Tertiary variable units code | 1 | CMD 3 | Cfg |
| 22 | byte | Quaternary variable units code | 1 | CMD 3 | Cfg |
| 23-30 | byte | Tag name | 8 | CMD 13 | Info |
| 31-46 | byte | Descriptor | 16 | CMD 13 | Info |
| 47-49 | byte | Tag/Descriptor data | 3 | CMD 13 | Info |
| 50-52 | byte | Transducer serial number | 3 | CMD 14 | Info |
| 53 | byte | Transducer limits and min span units code | 1 | CMD 14 | Info |
| 54 | byte | PV alarm selection code | 1 | CMD 15 | Info |
| 55 | byte | PV transfer function code | 1 | CMD 15 | Info |
| 56 | byte | PV upper and lower range value units code | 1 | CMD 15 | Info |
| 57 | byte | Write protection code | 1 | CMD 15 | Status |
| 58 | byte | Private label distributor code | 1 | CMD 15 | Info |
| 59 | byte | PV analog channel flag | 1 | CMD 15 | Info |
| | | **TOTAL BYTE COUNT** | **60** | | |
| | | **TOTAL WORD COUNT** | **30** | | |

| DB BYTE OFFSET | TYPE | DESCRIPTION | BYTE CNT | DATA SOURCE | USE OF DATA |
|---|---|---|---|---|---|
| 60-63 | float | Primary variable value | 4 | CMD 3 | Status |
| 64-67 | float | Secondary variable value | 4 | CMD 3 | Status |
| 68-71 | float | Tertiary variable value | 4 | CMD 3 | Status |
| 72-75 | float | Quaternary variable value | 4 | CMD 3 | Status |
| 76-79 | float | Upper transducer limit | 4 | CMD 14 | Cfg |
| 80-83 | float | Lower transducer limit | 4 | CMD 14 | Cfg |
| 84-87 | float | Minimum span | 4 | CMD 14 | Cfg |
| 88-91 | float | PV upper range value | 4 | CMD 15 | Cfg |
| 92-95 | float | PV lower range value | 4 | CMD 15 | Cfg |
| 96-99 | float | PV damping value (in seconds) | 4 | CMD 15 | Cfg |
| | | **TOTAL FLOAT BYTE COUNT** | **40** | | |
| | | **TOTAL FLOAT WORD COUNT** | **20** | | |

| | |
|---|---|
| **DB REGS/DEVICE** | **50** |
| **MAX DB REGS/CHANNEL FOR 15 DEVICES** | **750** |
| **MAX DB REGS FOR HART CARD** | **4000** |

The following table defines the auto-polling command status bits:

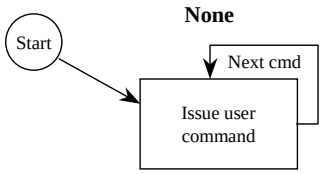
**AUTO-POLLING COMMAND STATUS BITS**

| Bit # | Description |
|---|---|
| 0 | Long Address Set (command 0 successful) |
| 1 | Command 13 successful (configuration) |
| 2 | Command 14 successful (configuration) |
| 3 | Command 15 successful (configuration) |
| 4 | Command 3 successful (data polling) |
| 5 | Reserved |
| 6 | Reserved |
| 7 | Reserved |

The following sections describe the modes of module operation.

## Auto-Poll Disabled Mode

If the auto-polling feature is disabled (Auto-Poll Code = N), the module functions as shown in the following diagram:

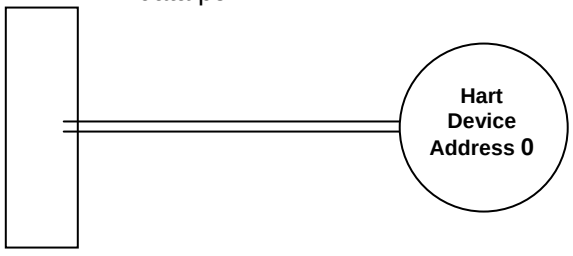



Only the user commands are executed and all data is placed in and sourced from the module's internal database. The user is responsible for constructing all commands to control and monitor the instruments attached to the channel.

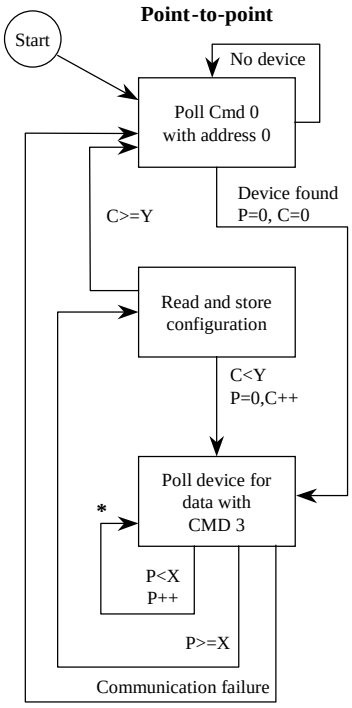
## Point-to-Point Mode

**Important:** If the HART device address is 0 you must configure the channel for Point-to-Point mode.

In point-to-point mode, the module only polls for a single instrument with a polling address of zero. When the instrument is found by the channel, it continuously polls for the data using command 3. Occasionally, it will poll for the configuration information for the device. This is accomplished with HART commands 13, 14 and 15. Less frequently, the channel will perform a HART command 0 request to see if any of the data for the instrument has changed. If communications is lost with the device, the module will try to establish communications with the device using command 0. If user commands are present and enabled, they will be executed before the next data poll.



When the point-to-point mode of auto-polling is enabled (Auto-Poll Code = P), the following diagram applies to the channel operation:

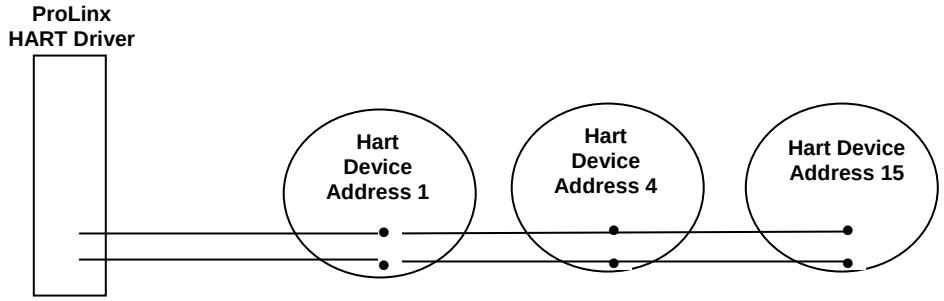



**Where**
**P, C = loop iterators**
**X = Poll Counter (internal fixed value)**
**Y = HART configuration counter (internal fixed value)**

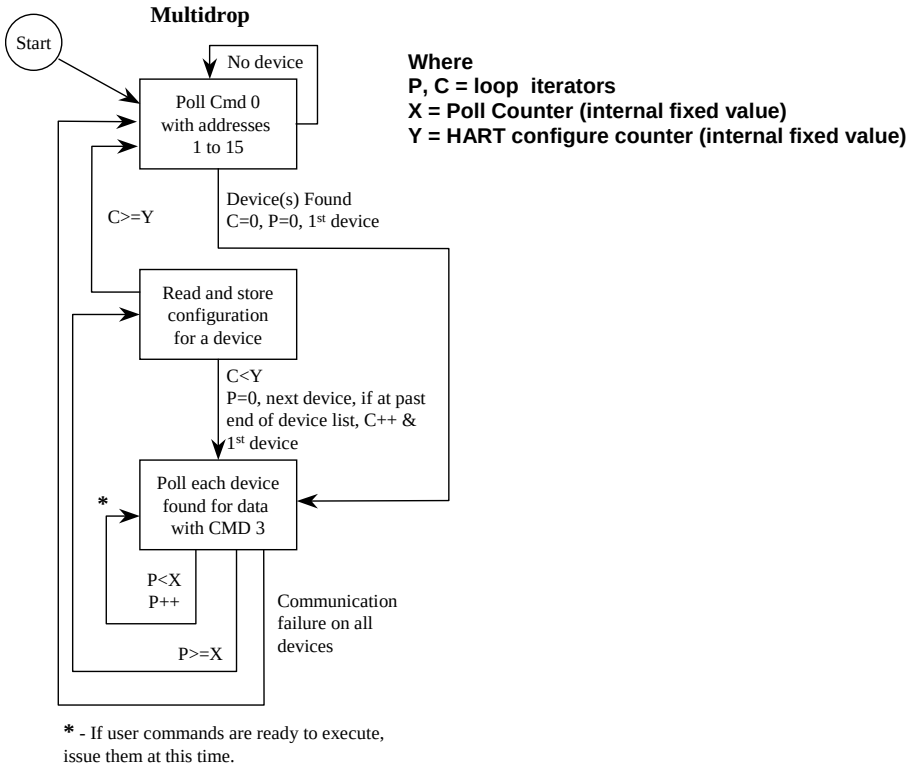
### 1.2.2.1 Multidrop Mode

**Important:** If the HART device address is between 1 and 15, you must configure the channel for multidrop mode.

If the unit is set for multidrop mode, the module will poll each unit attached to the channel starting with polling address 1. The parameter MAX DEVICE COUNT in the configuration determines the maximum slave address number to be polled in multidrop mode. For example, if the MAX DEVICE COUNT parameter is set to 3, the channel will poll for polling addresses 1, 2, and 3. It is important when assigning the device polling addresses for instruments on a channel to start with 1 and successfully increase the value by one until the last instrument is assigned an address. If you set the MAX DEVICE COUNT parameter to 2, and assign the two instruments addresses 1 and 15, the polling address 15 will never be used (only addresses 1 and 2). Therefore, the second instrument will never be polled and will be in communication failure. Up to 15 instruments can be assigned to a single channel (polling address 1 to 15).




If the module is configured for multidrop auto-polling (Auto-Poll Code = M), the following diagram applies:




In multidrop mode, the channel will poll for instruments 1 to the value set in the MAX DEVICE COUNT parameter. For each device found, it will continuously poll for data using command 3. After a certain number of iterations, the HART Driver will ready the configuration data for the next device (if there is one). Each time configuration information is acquired, it will be for a different slave device if more than 1 slave is recognized by the channel. Less frequently, the channel will poll for devices not found in the original search of the network using HART command 0. If communications is lost with all devices, the module will try to establish communications with the devices using command 0. If user commands are present and enabled, they will be executed after the data polling of the devices.

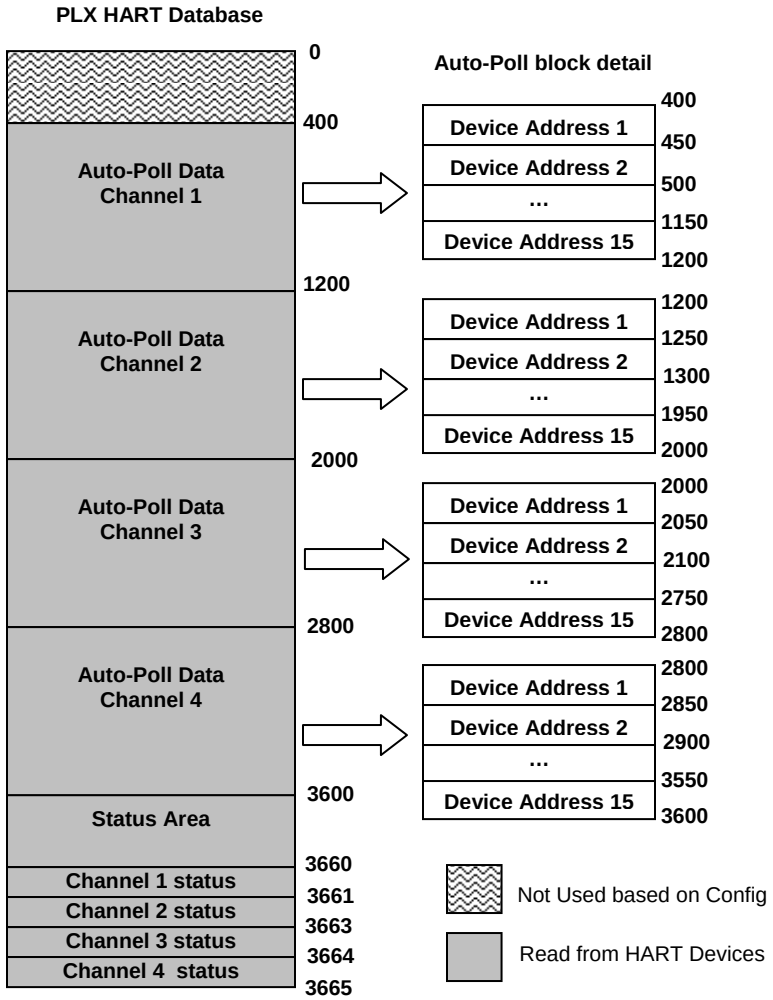
**Multidrop Mode Example:**
The intent is to show when using Multidrop mode how the Auto-Poll DB Address and Max Device Count parameters should be used.

If the configuration file sets the following parameters:

| Parameter | Value | Config File Section |
|---|---|---|
| DB Address Status | 3660 | [HART PORT 0] |
| Auto-Poll DB Address | 400 | |
| Max Device Count | 15 | |
| DB Address Status | 3661 | [HART PORT 1] |
| Auto-Poll DB Address | 1200 | |
| Max Device Count | 15 | |
| DB Address Status | 3662 | |

| Parameter | Value | Config File Section |
|---|---|---|
| Auto-Poll DB Address | 2000 | [HART PORT 2] |
| Max Device Count | 15 | |
| DB Address Status | 3663 | [HART PORT 3] |
| Auto-Poll DB Address | 2800 | |
| Max Device Count | 15 | |

The ProLinx HART Driver database will have the following structure.




The configuration file for this example shows that when the MAX Device Address parameter is configured as 15, the module will reserve 800 words for each channel (15 devices x 50 words each) even though not all devices may be present on the network.

### 1.2.3 HART Command List

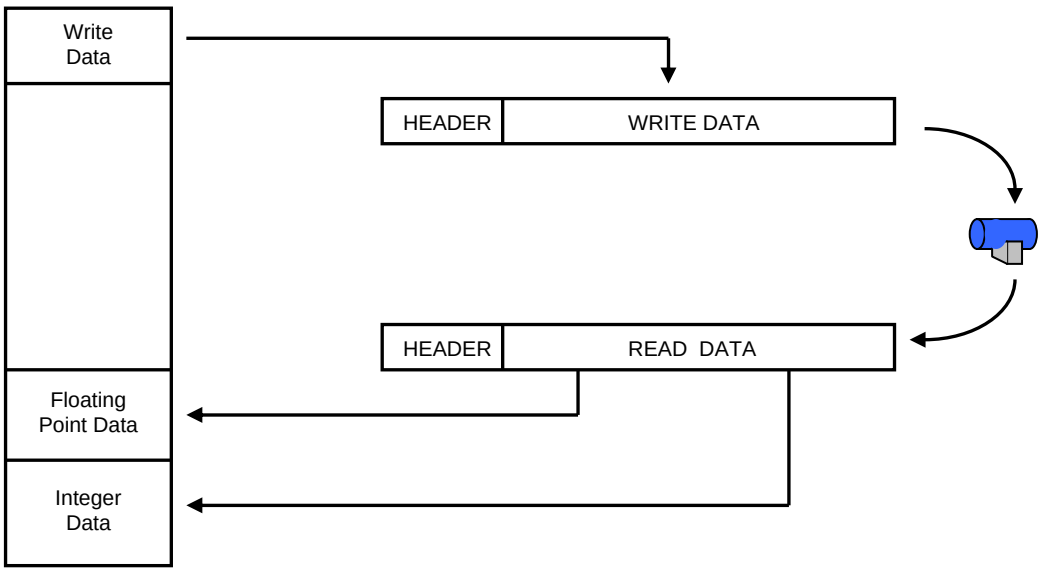
The HART Command List specifies the commands to be executed to the HART devices connected to a channel. A HART command can be seen as an outgoing message to the HART devices that provides Write Data for a specific command or a response message that carries process data (Read Data) back to the module. The Prolinx module supports three kinds of data blocks in the Universal and Common Practice commands. These data blocks are:

- Integers
- IEEE 754 Floating Point Numbers (32 bits)
- Packed ASCII character strings

The Packed ASCII character strings are unpacked and placed with the integers data block.

For all commands, it is possible to select where the Write Data comes from; it can be in the module's internal database or it can be configured as a fixed data block in the command.

For response messages from HART devices, it is possible to configure where the Floating Point Data and Integer Data will be placed in the module's internal database, but this is only possible for the Universal and Common Practice commands. In the case of Device Specific commands, all the Read Data is placed in the Integer data section. Look at Appendix A for all the supported HART commands.



# 2 Protocol Functional Specifications

## 2.1 HART Channel Specifications

| Type | Specifications |
|---|---|
| **General Parameters (Software Configurable)** | |
| Internal Database | 4000 registers (words) available |
| Communication parameters | Number of HART Preambles: 2 to 50<br>Enable Handheld: Y or N<br>Primary Master: Y or N |
| Configurable Parameters per Channel | Short Address Retries<br>Long Address Retries<br>Retries After Error<br>Poll Time After Error<br>Number of Commands<br>Slave List Error Pointer |
| HART Commands supported | Universal Command Set<br>00 Read Unique Identifier<br>01 Read Primary Variable<br>02 Read Current And Percent Of Range<br>03 Read Dynamic Variables<br>06 Write Polling Address<br>11 Read Unique Identifier Associated With Tag<br>12 Read Message<br>13 Read Tag Descriptor Date<br>14 Read PV Sensor Info<br>15 Read Output Information<br>16 Read Final Assembly Number<br>17 Write Message<br>18 Write Tag Descriptor Date<br>19 Write Final Assembly Number<br><br>Common Practice Command Set<br>33 Read Transmitter Variables<br>34 Write Damping Value<br>35 Write Range Values<br>36 Set Upper Range Value<br>37 Set Lower Range Value<br>38 Reset Configuration Changed Flag<br>39 EEprom Control<br>40 Enter Exit Fixed Current Mode<br>41 Perform Transmitter Self Test<br>42 Perform Master Reset<br>43 Set Pv Zero<br>44 Write Pv Units<br>45 Trim Dac Zero<br>46 Trim Dac Gain<br>47 Write Transfer Function<br>48 Read Additional Transmitter Status<br>49 Write Pv Sensor Serial Number<br>50 Read Dynamic Variable Assignments<br>51 Write Dynamic Variable Assignments<br>52 Set Transmitter Variable Zero<br>53 Write Transmitter Variable Units<br>54 Read Transmitter Variable Information<br>55 Write Transmitter Variable Damping Value<br>56 Write Transmitter Variable Sensor Serial Number |

| | |
|---|---|
| | 57 Read Unit Tag Descriptor Date<br>58 Write Unit Tag Descriptor Date<br>59 Write Number Of Response Preambles<br>108 Write Burst Mode Command Number<br>109 Burst Mode Control<br>110 Read All Dynamic Variables |
| Command List | Up to 100 command per channel, each fully configurable for function, slave address, register to/from addressing and word count for Floating Point or Integer Data |
| Status Data | Error codes returned by the HART protocol available on an individual command basis. In addition, a slave status list is maintained per active channel. |
| Polling of command list | User configurable polling of commands, including disabled, continuous, on change of data (write only) and dynamically user or automatic enabled. |

# 3 HART Protocol Specific Configuration File

Following is an example view of the HART portion of the configuration file showing only one HART channel. This example should serve only to give the programmer an idea of how a CFG file is structured. Complete configuration files are shipped on each unit and are available off the web site for each of the products. These files can serve as an excellent starting point for any project.

```
# This section is used to define the configuration for the HART channel
#
#
[HART Port 0]
Enabled                 : Yes    #Yes=Use port, No=Do not use port
Preambles               : 5      #Number of Preambles to be sent to HART Network
Enable Handheld         : Yes    #Yes=Let the use of HART Hand Held No=Don't
Primary Master          : Yes    #Yes=The Master is Primary No=Secondary
Short Address Retries   : 3      #Number of retries with short address
Long Address Retries    : 3      #Number of retries with long address
Retries After Error     : 0      #Number of retries after error
Poll Time After Error   : 10     #Poll Time After Timeout of a Device
DB Address Status       : 100    #Database Address to place device status
Command Count           : 1      #Number of Commands in Command Table


[HART Port 0 Commands]
# The file contains examples for a pressure transducer.
START
#      1       2       3      4    5       6    7       8     9   10   11      12       13      14     15   16    17
#        Float DB     Poll Word  Swap   Short Func Int. DB Word  Swap Use  Enable DB Done DB Write DB Swap Byte  Fixed
# Enable Address  Interval Count Code Address Code Address Count Code Long Address   Address Address  Code Count Data
    3        0        0      0    2       2    44      0      2    0    1      -1        9       -1      0     1   04
    5        0        0      0    2       2    44     10      2    0    1       9       19       -1      0     1   04
    5       22        0      6    2       2   110     28      1    0    1      19       29       -1      0     0
    5        0        0      0    2       2    44     30      2    0    1      29       39       -1      0     1   05
    5       40        0      8    2       2     3     48      1    0    1      39        9       -1      0     0
    1       50        0      4    2       3     2     54      2    0    1      -1       59       -1      0     0
    1       60        0      0    2       3    12     60     14    0    1      -1       -1       -1      0     0
END
```

# 4 CFG File: [HART PORT x] Section

The [HART PORT 0], [HART PORT 1], [HART PORT 2], [HART PORT 3] [HART PORT 4], [HART PORT 5], [HART PORT 6] and [HART PORT 7] sections of the **DFCMHART.CFG** file are used to set the HART channel communication parameters, define the protocol specifics and set the command list parameters. The parameters are the same for all 8 sections. The command list for each HART channel is entered in a different section in the file. The table below lists the parameters defined in this section:

| [SECTION]/Item | Range | Description |
|---|---|---|
| [HART PORT 0]<br>[HART PORT 1]<br>[HART PORT 2]<br>[HART PORT 3]<br>[HART PORT 4]<br>[HART PORT 5]<br>[HART PORT 6]<br>[HART PORT 7] | | Configuration Header for Port 0<br>Configuration Header for Port 1<br>Configuration Header for Port 2<br>Configuration Header for Port 3<br>Configuration Header for Port 4<br>Configuration Header for Port 5<br>Configuration Header for Port 6<br>Configuration Header for Port 7 |
| Enabled: | Yes or No | This flag specifies if the channel on the module will be utilized. If the parameter is set to No, the channel will not be used. If the parameter is set to Yes, the channel will be used supporting the HART protocol. |
| Preambles: | 2 to 50 | This parameter defines the number of 0xFF characters (preambles) that will be placed at the start of each HART frames. |
| Enable Handheld: | Yes or No | This parameter enables or disables the use of handheld in the channel. If the parameter is set to No, the ProLinx module will not let other master to take data from the channel. If set to Yes, it is possible to place other master in channel. With this option in N the best throughput is achieved on the channel. |
| Primary Master: | Yes or No | This parameter specifies the Link Quiet / Slave Timeout. If set to Yes, the selected Link Quiet / Slave Timeout is 305 mSec which corresponds to the Primary Master. If set to No, then this time is 380 mSec which corresponds to the Secondary Master. |
| Short Address Retries: | 0 to 10 | This parameter specifies the number of times a command addressed with short address will be retried if it fails. |
| Long Address Retries: | 0 to 10 | This parameter specifies the number of times a command addressed with long address will be retried if it fails. |
| Retries After Error: | 0 to 10 | This parameter specifies the number of times a command will be retried if it fails and the device to which the command is addressed is in error. |
| Poll Time After Error: | 0 to 65535 | This parameter sets the time to wait to execute a command again if it failed to respond for the retries number. This facility can improve communication throughput on the HART network. |
| DB Address Status | 0-3999 | This is the database address to use to place the device status. |
| Command Count: | 0 to 100 | This parameter specifies the number of commands to be processed for the port. |

# 5 CFG File: [HART PORT x COMMANDS] Section

The [HART PORT 0 COMMANDS], [HART PORT 1 COMMANDS], [HART PORT 2 COMMANDS], [HART PORT 3 COMMANDS], [HART PORT 4 COMMANDS], [HART PORT 5 COMMANDS], [HART PORT 6 COMMANDS] and [HART PORT 7 COMMANDS] sections of the CFG file are used to set the HART channel command lists. These lists are used to poll slave devices attached to the HART channels. The module supports numerous commands.

The command list is formatted differently than the other sections of the configuration file. Commands are present in a block between the labels **START** and **END**. These labels are used to inform the program where the list resides. The module's program will parse all commands after the **START** label until it reaches the **END** label or until the command count entered for the port is reached.

## 5.1 Command List Overview

In order to interface the ProLinx module with HART slave devices, the user must construct a command list. The commands in the list specify the slave device to be addressed, the function to be performed (read or write), and the registers in the internal database to be associated with the device data. There is a separate command list for each HART channel, with up to 100 commands allowed per channel. The command list is processed from top (command #0) to bottom. A poll interval parameter is associated with each command to specify a minimum delay time in seconds between the issuance of a command. If the user specifies a value of 10 for the parameter, the command will be executed no more frequently than every 10 seconds.

Write commands have a special feature, as they can be set to execute only if the data in the write command changes. If the register data values in the command have not changed since the command was last issued, the command will not be executed. If the data in the command has changed since the command was last issued, the command will be executed. Use of this feature can lighten the load on the HART network. In order to implement this feature, set the enable code for the command to a value of 2.

The module supports all the Universal and Common Practice commands and also a Device Specific command is supported without any translation of the data.

## 5.2 Commands Supported by the Module

The format of each command in the list is independent on the function being executed. All parameters in the command table must be entered. Only one parameter is optional, the Fixed Data field, which contains data to be sent to a HART device. The tables below list the functions supported by the module:

### HART Universal Commands Set

| Command | Definition |
|---|---|
| 00 | Read Unique Identifier |
| 01 | Read Primary Variable |
| 02 | Read Current And Percent Of Range |
| 03 | Read Dynamic Variables |
| 06 | Write Polling Address |
| 11 | Read Unique Identifier Associated With Tag |
| 12 | Read Message |
| 13 | Read Tag Descriptor Date |
| 14 | Read PV Sensor Info |
| 15 | Read Output Information |
| 16 | Read Final Assembly Number |
| 17 | Write Message |
| 18 | Write Tag Descriptor Date |
| 19 | Write Final Assembly Number |

### HART Common Practice Commands Set

| Command | Definition |
|---|---|
| 33 | Read Transmitter Variables |
| 34 | Write Damping Value |
| 35 | Write Range Values |
| 36 | Set Upper Range Value |
| 37 | Set Lower Range Value |
| 38 | Reset Configuration Changed Flag |
| 39 | EEPROM Control |
| 40 | Enter Exit Fixed Current Mode |
| 41 | Perform Transmitter Self Test |
| 42 | Perform Master Reset |
| 43 | Set PV Zero |
| 44 | Write PV Units |
| 45 | Trim DAC Zero |
| 46 | Trim DAC Gain |
| 47 | Write Transfer Function |
| 48 | Read Additional Transmitter Status |
| 49 | Write PV Sensor Serial Number |
| 50 | Read Dynamic Variable Assignments |
| 51 | Write Dynamic Variable Assignments |
| 52 | Set Transmitter Variable Zero |
| 53 | Write Transmitter Variable Units |
| 54 | Read Transmitter Variable Information |
| 55 | Write Transmitter Variable Damping Value |
| 56 | Write Transmitter Variable Sensor Serial Number |
| 57 | Read Unit Tag Descriptor Date |
| 58 | Write Unit Tag Descriptor Date |
| 59 | Write Number Of Response Preambles |
| 108 | Write Burst Mode Command Number |
| 109 | Burst Mode Control |
| 110 | Read All Dynamic Variables |

## 5.3 HART Command Entry Formats

**Appendix Reference**

Refer to the Appendix for a complete discussion of the HART commands supported by the module and of the structure and content of the data returned for each command.

An example of a command list section of the CFG file is displayed below:

```
[HART Port 0 Commands]
# The file contains examples for a pressure transducer.
START
#      1       2       3      4    5       6    7       8     9   10   11      12       13      14     15   16    17
#        Float DB     Poll Word  Swap   Short Func Int. DB Word  Swap Use  Enable DB Done DB Write DB Swap Byte  Fixed
# Enable Address  Interval Count Code Address Code Address Count Code Long Address   Address Address  Code Count Data
    3        0       0      0    2       2    44      0      2    0    1      -1       9       -1      0     1   04
    5        0       0      0    2       2    44     10      2    0    1       9       19       -1      0     1   04
    5       22       0      6    2       2   110     28      1    0    1      19       29       -1      0     0
    5        0       0      0    2       2    44     30      2    0    1      29       39       -1      0     1   05
    5       40       0      8    2       2     3     48      1    0    1      39        9       -1      0     0
    1       50       0      4    2       3     2     54      2    0    1      -1       59       -1      0     0
    1       60       0      0    2       3    12     60     14    0    1      -1       -1       -1      0     0
END
```

Each parameter is discussed below:

| Command Parameter | Range | Description |
|---|---|---|
| Enable | 0 ,1,2,3,4,5 | This field is used to define whether or not the command is to be executed and under what conditions.<br>If the parameter is set to 0, the command is disabled and will not be executed in the normal polling sequence.<br>Setting the parameter to a value of 1 for the command causes the command to be executed each scan of the command list if the Poll Interval Time is set to zero. If the Poll Interval time is set, the command will be executed, when the interval timer expires.<br>If the parameter is set to 2, the command will execute only if the internal data associated with the command changes. This value is valid only when there is a specified “Write DB Address” (see below) with a non zero bytecount for write commands.<br>If the parameter is set to 3, the command will execute only once on startup of the ProLinx module after a restart. This is mainly used for configuration of HART devices on startup.<br>Setting this parameter to 4, places the command in enabled mode. This option is valid only if there is a specified “Enabled DB Address” (see below). If the Virtual Database word specified in “Enabled DB Address” has “-1” the command will be executed otherwise it will not.<br>Setting this parameter to 5, places the command in one shot enabled mode. This option is valid only if there is a specified “Enabled DB Address” (see below). If the Virtual Database word specified in “Enabled DB Address” has “-1” the command will be executed otherwise it will not. When the command has been successful the Virtual Database word specified in “Enabled DB Address” will be written with “0”, so the command will be executed only once. |
| Float DB Address | 0 to 3999 | This field specifies the internal database register where the floating point values returned by the command will be placed.<br>Register addresses specified for commands must reside in the range |

| Command Parameter | Range | Description |
|---|---|---|
| | | specified by the Maximum Register parameter under the [MODBUS MASTER] section. |
| Poll Interval | 0 to 65535 | This parameter specifies the minimum interval to execute continuous commands (Enable code of 1). The parameter is entered in units of seconds. Therefore, if a value of 10 is entered for a command, the command will execute no more frequently than every 10 seconds. |

| | | |
|---|---|---|
| Word Count | -1 to 125 | This parameter specifies the number of words from the floating point data returned by a HART command that will be placed on the Virtual Database. If this parameter is 0 no data will be written to the Database. If this parameter is –1 then all the floating point data will be written in the integer block of data.<br>Special care should be taken with this number, because is a word count and a floating point value is 2 words long. For example if you execute a HART command 3 which takes 5 floating point values from the device, you should place a word count of 10 words. |
| Swap Code | 0,1,2,3 | This parameter is used to define if the data received from the HART slave is to be ordered differently than received from the slave device. This parameter is helpful when dealing with floating-point or other multi-register values, as there is no standard method of storage of these data types in slave devices. This parameter can be set to order the register data received in an order useful by other applications. The table below defines the values and their associated operations:<br><br>Swap Code<br>Description<br><br>0<br>None – No Change is made in the byte ordering<br><br>1<br>Words – The words are swapped<br><br>2<br>Words & Bytes – The words are swapped then the bytes in each word are swapped<br><br>3<br>Bytes – The bytes in each word are swapped |
| Short Address | 1 to 15 | This parameter is used to specify the HART slave node address on the network to be considered. Values of 1 to 15 are permitted. If the device to be addressed only accepts long address, then the parameter “Use Long” should be selected so the module can ask for the long address with the short one and then execute the command. |
| Function Code | 0 to 255 | This parameter is used to specify the HART function to be executed. Any HART function can be executed, even device specific ones, but only supported commands will return formatted data and classified in floating point data and integer data. |
| Int. DB Address | 0 to 3999 | This field specifies the internal database register where the integer or packed ASCII string values returned by the command will be placed.<br>If the Floating Point Word Count parameter is “–1”, then all the data returned by the HART command will be placed in this address without any formatting.<br>Register addresses specified for commands must reside in the range specified by the Maximum Register parameter under the [MODBUS MASTER] section. |
| Word Count | 0 to 125 | This parameter specifies the number of words from the integer or packed ASCII string data returned by a HART command that will be placed on the Virtual Database. If this parameter is “0”, no data will be written to the Database. |
| Swap Code | 0,1,2,3 | This parameter is used to define if the data received from the HART slave is to be ordered differently than received from the slave device. Its format is the same that the specified for the Floating Point Data Swap Code (see above). |

| | | |
|---|---|---|
| Use Long | 0 or 1 | This parameter is used to define if the command will be executed with short or long address. If the value is “0”, then the configured command will be executed using the Short Address specified. If the value is “1” then the specified Short Address will be used only to ask for the long address and that will be used to execute the configured command. |
| Enable DB Address | 0 to 3999 | This field specifies the internal database register to be used to enable the execution of a command. This parameter is only used if “Enable” is “4” or “5”. If the value of this database register is “-1”, then the command will be executed, otherwise it will not. If the “Enable” value is “5”, then after the successful execution of the command this value will become “0” |
| Done DB Address | -1 to 3999 | This field specifies the internal database register to be used to signal the successful execution of a command. When a command is successfully executed a “-1” is written to this register. This “Done DB Address” can be shared with the “Enable DB Address” of another command to do a chained command execution. |
| Write DB Address | -1 to 3999 | This field specifies the internal database register to be as a source of data for HART command which includes data. It is possible to include data with every HART command, but it depends of the command and of the device if it will accept this data.<br>If this value is “-1”, it indicates that there will not be data with the command or that it will not come from database. |
| Swap Code | 0,1,2,3 | This parameter is used to define if the data to be sent to a HART slave is to be ordered differently than data that resides in the Virtual Database. Its format is the same that the specified for the Floating Point Data Swap Code (see above). |
| Byte Count | 0 to 250 | This parameter specifies the number of bytes to be sent to a HART device in the command. If the command has no data then this value should be “0”. If the value of this field is different of “0” and “Write DB Address” is different of “-1” then the data for the command will be taken from the Virtual Database. If the value of this field is different of “0” and “Write DB Address” is “-1” then the data for the command will be from the “Fixed Data” field for the command. |
| Fixed Data | Up to 250 HEX values separated by space | This parameter is a string of HEX values to be sent with the HART command. There should be at least the number of bytes specified in the “Byte Count” parameter. The bytes should be written in hexadecimal format and separated by a space. This data will be sent if the Parameter “Write DB Address” is “-1” and “Byte Count” is greater than “0”, and it will be sent in the same order that they are written. |

# 6 Error/Status Data

The module error/status data areas are discussed in this section. The module contains three areas related to this data. The user defines the location of these data sets in the virtual database of the module. The error/status data contains module data, the six command error list data sets contain the errors associated with each respective command list and the four slave status list tables contain the communication status of each port's slave devices.

## 6.1 Viewing Error and Status Data

The following sections list the register addresses that contain error and status data. You use the Database View option to view the contents of these registers. Refer to the *ProLinx Communication Gateways, Inc. Startup Guide* for detailed information on viewing error and status registers.

## 6.2 Configuration Error Word Table

**HART CONFIGURATION ERROR WORD**

| Bit | Code | Description |
|---|---|---|
| 0 | 0x0001 | Enabled not set to Y or N |
| 1 | 0x0002 | Enable Handheld not set to Y or N |
| 2 | 0x0004 | Primary Master not set to Y or N |
| 3 | 0x0008 | Invalid Preambles (1 to 50) |
| 4 | 0x0010 | Invalid Short Address Retries (0 to 50) |
| 5 | 0x0020 | Invalid Long Address Retries (0 to 50) |
| 6 | 0x0040 | Invalid Retries After Error (0 to 50) |
| 7 | 0x0080 | Invalid Poll Time After Error (0 to 10000) |
| 8 | 0x0100 | Invalid DB Address Status |
| 9 | 0x0200 | Invalid Command Count |
| 10 | 0x0400 | Memory Error in allocating commands |
| 11 | 0x0800 | Memory Error in allocating command fixed data |
| 12 | 0x1000 | Memory Error in allocating TX/RX buffers |
| 13 | 0x2000 | HART Board not found |
| 14 | 0x4000 | Cannot initialize HART channel |
| 15 | 0x8000 | reserved |

## 6.3 Port 0 General Status Data

| Status Register | Description |
|---|---|
| 10700 | State |
| 10701 | Comm State |
| 10702 | Device Status (bit mapped) |
| 10703 | Device Poll List (bit mapped) |
| 10704 | Device With Long Address (bit mapped) |
| 10705 | Current Command |
| 10706 | Command Request Count |
| 10707 | Command Response Count |
| 10708 | Configuration Error Word |
| 10709 | Current Error Code |
| 10710 | Last Error Code |
| 10711<br>…<br>…<br>10799 | No Valid Data |

## 6.4 Port 1 General Status Data

| Status Register | Description |
|---|---|
| 10900 | State |
| 10901 | Comm State |
| 10902 | Device Status (bit mapped) |
| 10903 | Device Poll List (bit mapped) |
| 10904 | Device With Long Address (bit mapped) |
| 10905 | Current Command |
| 10906 | Command Request Count |
| 10907 | Command Response Count |
| 10908 | Configuration Error Word |
| 10909 | Current Error Code |
| 10910 | Last Error Code |
| 10911<br>…<br>…<br>10999 | No Valid Data |

### 6.4.1 General Status Data (Ports 2 through 7)

The following table lists the register address ranges for ports 2 through 7. The register descriptions are the same as listed in the previous table.

| Port | Address Range |
|---|---|
| 2 | 11100 - 11299 |
| 3 | 11300 - 11499 |
| 4 | 11500 - 11699 |
| 5 | 11700 - 11899 |
| 6 | 11900 - 12099 |
| 7 | 12100 - 12299 |

# 7 Error/Status Data Structure

The error/status data table is located at the virtual address assigned by the user. If the address is set to -1 or the frequency parameter is set to 0, the data will not be placed in the database. It will only be available through the Configuration/Debug Port. If valid address and frequency values are assigned, the module will update the data area.

The data area will be initialized with zeros whenever the processor is initialized. This occurs during a cold-start (power-on), reset (reset push-button pressed) or a warm-boot operation (commanded or loading of new configuration).

**4201-DFCM-HART (Single DFCM port – Four HART channels)**

The error/status data area is a 44-word register block, the structure of which is shown below:

| WORD | DESCRIPTION |
|---|---|
| **DF1 PORT 0** | |
| 9 | Number of Command Requests |
| 10 | Number of Command Responses |
| 11 | Number of Command Errors |
| 12 | Number of Requests |
| 13 | Number of Responses |
| 14 | Number of Errors Received |
| 15 | Number of Errors Sent |
| **HART CHANNEL 0** | |
| 16 | Number of Command Requests |
| 17 | Number of Command Responses |
| 18 | Devices in Poll |
| 19 | Devices Status |
| 20 | Configuration Error |
| 21 | Current Error |
| 22 | Last Error |
| **HART CHANNEL 1** | |
| 23 | Number of Command Requests |
| 24 | Number of Command Responses |
| 25 | Devices in Poll |
| 26 | Devices Status |
| 27 | Configuration Error |
| 28 | Current Error |
| 29 | Last Error |
| **HART CHANNEL 2** | |
| 30 | Number of Command Requests |
| 31 | Number of Command Responses |
| 32 | Devices in Poll |
| 33 | Devices Status |
| 34 | Configuration Error |
| 35 | Current Error |
| 36 | Last Error |
| **HART CHANNEL 3** | |
| 37 | Number of Command Requests |
| 38 | Number of Command Responses |
| 39 | Devices in Poll |
| 40 | Devices Status |
| 41 | Configuration Error |
| 42 | Current Error |
| 43 | Last Error |

## 7.1 HART Command List Error Data

Each command in the command list for each HART channel has a word value for a status/error code. This error data list can be read using the Configuration/Debug Port and can be placed in the module's internal database. Accessing the Debug capabilities of the module is accomplished easily by connecting a PC to the Debug port and loading a terminal program. If using a module with hardware version 1, download PSTerm (see *'ProLinx Communication Gateways, Inc. Startup Guide'*). If using hardware version 2 (indicated with a 'V2' sticker on the back of the module) any terminal program can be used. This status/error code is the first word of the Integer Data Block returned from every HART command executed. This word has information about the execution of the command by the HART device.

Refer to later sections containing Error Codes to interpret the status/error codes present in the integer data area.

### 7.1.1 Port 0 Command List Errors

| Status Register | Description |
|---|---|
| 10800<br>…<br>…<br>10899 | Command List Errors |

### 7.1.2 Port 1 Command List Errors

Port 1 Command List errors are found in registers 11000 through 11099.

## 7.2 Slave List Status

The configuration parameter "DB Address Status" is used to define the register address in the virtual database where the status data for each HART channel will be placed. This word has one bit for each HART device and if this bit is in "1" it means that the corresponding HART device is not answering to the poll.

| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| HART DEVICE | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Res |

The bit 0 of the "Slave List Status" is reserved for future use and is always 0.

# 8 Module Error Codes

The module error codes are listed in this section. Error codes returned from the HART device are placed at the first word of the integer data block in the Virtual Database. The error codes are formatted in the word as follows: If the most-significant bit of the word contains “1”, then the most significant byte has a communication error code. The least-significant byte of the word will contain “0”.

If the most-significant bit of the word contains “0”, the most significant byte contains a command error code. In this case, the least-significant byte contains a device malfunction information.

If this word value is “-1” (or 0xFFFF), it means that the command timed out.

Use the error codes returned for each command in the list to determine the success or failure of the command. If the command fails, use the error code to determine the cause of failure.

## 8.1 Protocol Error Codes

These are error codes that are part of the HART protocol. The standard HART error codes are shown in the tables below:

Error Code Word

| First Byte | | | | | | | | Second Byte | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| BIT 7 OF FIRST BYTE = 1<br>COMMUNICATION ERROR | |
|---|---|
| **FIRST BYTE** | |
| BIT 6 | Parity Error |
| BIT 5 | Overrun Error |
| BIT 4 | Framing Error |
| BIT 3 | Checksum Error |
| BIT 2 | Reserved |
| BIT 1 | RX Buffer Overflow |
| BIT 0 | Undefined |
| **SECOND BYTE** | |
| BIT 0 to BIT 7 | All 0 |

| BIT 7 OF FIRST BYTE = 0<br>COMMAND ERROR | |
|---|---|
| **FIRST BYTE** | |
| Bits 6 to 0 (not bit-mapped): | |
| 0 | No Error |
| 1 | Undefined |
| 2 | Invalid Selection |
| 3 | Passed Parameter Too Large |
| 4 | Passed Parameter Too Small |
| 5 | Too Few Data Bytes Received |
| 6 | Transmitter-Specific Command Error |
| 7 | In Write-Protect Mode |
| 8-15 | Command Specific Errors (see Below) |
| 16 | Access Restricted |
| 32 | Device is Busy |
| 64 | Command not Implemented |
| **SECOND BYTE** | |
| BIT 7 | Device Malfunction |
| BIT 6 | Configuration Changed |
| BIT 5 | Cold Start |
| BIT 4 | Unused |
| BIT 3 | Output Current Fixed |
| BIT 2 | Analog Output Saturated |
| BIT 1 | Variable (Not Primary) Out of Limits |
| BIT 0 | Primary Variable Out of Limits |

| COMMAND SPECIFIC ERRORS | |
|---|---|
| 8 | Update Failed<br>Update In Progress<br>Set to Nearest Possible Value |
| 9 | Applied Process Too High<br>Lower Range Value Too High<br>Not in Fixed Current Mode |
| 10 | Applied Process Too Low<br>Lower Range Value Too Low<br>MultiDrop Not Supported |
| 11 | In MultiDrop Mode<br>Invalid Transmitter Variable Code<br>Upper Range Value Too High |
| 12 | Invalid Unit Code<br>Upper Range Value Too Low |
| 13 | Both Range Values Out of Limits |
| 14 | Pushed Upper Range Value Over Limit<br>Span Too Small |

# Appendix A: HART Command Support

**HART UNIVERSAL COMMANDS**

## COMMAND 00: Read Unique Identifier

**Description:**

This command gets the long address of the HART device plus other manufacturer information like Manufacturer Id, Device Type Code, Software Revision, Hardware Revision, etc.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Constant “254” | Manufacturer Identification Code |
| 2 | Manufacturer Device Type Code | Number of Preambles |
| 3 | Universal Command Revision | Transmitter Specific Command Revision |
| 4 | Software Revision | Hardware Revision |
| 5 | Device Function Flags | Device ID Number 1 |
| 6 | Device ID Number 2 | Device ID Number 3 |

## COMMAND 01: Read Primary Variable

**Description:**

This command gets the device Primary Variable and the Primary Variable Units

**Write Parameters:**

NONE

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Primary Variable Value | |
| 1 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Primary Variable Units Code | 0 |

## COMMAND 02: Read Current And Percent Of Range

**Description:**

This command gets the current of the loop that is forced by the HART device and the Percent of Range of the Current.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Current (mA) | |
| 1 | | |
| 2 | Percent of Range | |
| 3 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 03: Read Dynamic Variables

**Description:**

This command gets the current and four (predefined) dynamic Variables.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Current (mA) | |
| 1 | | |
| 2 | Primary Variable | |
| 3 | | |
| 4 | Second Variable | |
| 5 | | |
| 6 | Third Variable | |
| 7 | | |
| 8 | Fourth Variable | |
| 9 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Primary Variable Units Code | Second Variable Units Code |
| 2 | Third Variable Units Code | Fourth Variable Units Code |

## COMMAND 06: Write Polling Address

**Description:**

This command sets the polling address of a HART device. Extreme care should be taken when you use this command because you can loose the communication with the device.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Polling Address | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Polling Address | 0 |

## COMMAND 12: Read Message

**Description:**

This command reads an ASCII message contained in the HART Device and written by the Write Message command 17.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | Message ASCII Character 0 | Message ASCII Character 1 |
| **2** | Message ASCII Character 2 | Message ASCII Character 3 |
| . . . | . . . | |
| **14** | Message ASCII Character 28 | Message ASCII Character 29 |
| **15** | Message ASCII Character 30 | Message ASCII Character 31 |

## COMMAND 13: Read Tag, Descriptor and Date

**Description:**

This command reads an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | TAG ASCII Character 0 | TAG ASCII Character 1 |
| **2** | TAG ASCII Character 2 | TAG ASCII Character 3 |
| **3** | TAG ASCII Character 4 | TAG ASCII Character 5 |
| **4** | TAG ASCII Character 6 | TAG ASCII Character 7 |
| **5** | Descriptor ASCII Character 0 | Message ASCII Character 1 |
| . . . | | . . . |
| **12** | Descriptor ASCII Character 14 | Message ASCII Character 15 |
| **13** | Date | Date |
| **14** | Date | 0 |

## COMMAND 14: Read PV Sensor Info

**Description:**

This command gets information about the Primary Variable sensor, like limits and span.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Upper Sensor Limit | |
| 1 | | |
| 2 | Lower Sensor Limit | |
| 3 | | |
| 4 | Minimum Span | |
| 5 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Sensor Serial Number 0 | Sensor Serial Number 1 |
| 2 | Sensor Serial Number 2 | Unit Codes for Sensor Limits and Span |

## COMMAND 15: Read Output Information

**Description:**

This command gets information about the Primary Variable Output Information.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Upper Range Value | |
| 1 | | |
| 2 | Lower Range Value | |
| 3 | | |
| 4 | Damping Value (Sec) | |
| 5 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Alarm Select Code | Transfer Function Code |
| 2 | Primary Variable Range Units Code | Write-Protect Code |
| 2 | Private-Label Distributor Code | 0 |

## COMMAND 16: Read Final Assembly Number

**Description:**

This command reads the final assembly number of the HART device.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | Final Assembly Number 0 | Final Assembly Number 1 |
| **2** | Final Assembly Number 2 | 0 |

## COMMAND 17: Write Message

**Description:**

This command writes an ASCII message contained in the HART Device and that can be read with command 12.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Packed ASCII Message Byte 0 | Packed ASCII Message Byte 1 |
| 1 | Packed ASCII Message Byte 2 | Packed ASCII Message Byte 3 |
| . . . | . . . | |
| 11 | Packed ASCII Message Byte 22 | Packed ASCII Message Byte 23 |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 18: Write Tag, Descriptor and Date

**Description:**

This command writes an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | Packed ASCII TAG Byte 0 | Packed ASCII TAG Byte 1 |
| **1** | Packed ASCII TAG Byte 2 | Packed ASCII TAG Byte 3 |
| **2** | Packed ASCII TAG Byte 4 | Packed ASCII TAG Byte 5 |
| **3** | Packed ASCII Descriptor Byte 0 | Packed ASCII Descriptor Byte 1 |
| **.<br>.<br>.** | | **.<br>.<br>.** |
| **8** | Packed ASCII Descriptor Byte 10 | Packed ASCII Descriptor Byte 11 |
| **9** | Date | Date |
| **10** | Date | 0 |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | TAG ASCII Character 0 | TAG ASCII Character 1 |
| **2** | TAG ASCII Character 2 | TAG ASCII Character 3 |
| **3** | TAG ASCII Character 4 | TAG ASCII Character 5 |
| **4** | TAG ASCII Character 6 | TAG ASCII Character 7 |
| **5** | Descriptor ASCII Character 0 | Message ASCII Character 1 |
| **.<br>.<br>.** | | **.<br>.<br>.** |
| **12** | Descriptor ASCII Character 14 | Message ASCII Character 15 |
| **13** | Date | Date |
| **14** | Date | 0 |

## COMMAND 19: Write Final Assembly Number

**Description:**

This command writes the final assembly number of the HART device.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | Final Assembly Number 0 | Final Assembly Number 1 |
| **1** | Final Assembly Number 2 | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | Final Assembly Number 0 | Final Assembly Number 1 |
| **2** | Final Assembly Number 2 | 0 |

# COMMAND 33: Read Transmitter Variables

**Description:**

This command gets four user selected dynamic Variables.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Code For Slot 0 | Transmitter Variable Code For Slot 1 |
| 1 | Transmitter Variable Code For Slot 2 | Transmitter Variable Code For Slot 3 |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Variable for Slot 0 | |
| 1 | | |
| 2 | Variable for Slot 0 | |
| 3 | | |
| 4 | Variable for Slot 2 | |
| 5 | | |
| 6 | Variable for Slot 3 | |
| 7 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Transmitter Variable Code For Slot 0 | Units Code for Slot 0 Variable |
| 2 | Transmitter Variable Code For Slot 1 | Units Code for Slot 1 Variable |
| 3 | Transmitter Variable Code For Slot 2 | Units Code for Slot 2 Variable |
| 4 | Transmitter Variable Code For Slot 3 | Units Code for Slot 3 Variable |

## COMMAND 34: Write Damping Value

**Description:**

This command writes the damping value of a HART device.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point Damping Value (Sec) | |
| 1 | | |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point Damping Value (Sec) | |
| 1 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 35: Write Range Values

**Description:**

This command writes the Upper and Lower range of the Primary Variable

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | Range Units Code | Floating Point Upper Range Value (Byte 0) |
| **1** | Floating Point Upper Range Value (Byte 1) | Floating Point Upper Range Value (Byte 2) |
| **2** | Floating Point Upper Range Value (Byte 3) | Floating Point Lower Range Value (Byte 0) |
| **3** | Floating Point Lower Range Value (Byte 1) | Floating Point Lower Range Value (Byte 2) |
| **4** | Floating Point Lower Range Value (Byte 3) | |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | Upper Range Value | |
| **1** | | |
| **2** | Lower Range Value | |
| **3** | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | Range Units Code | 0 |

## COMMAND 36: Set Upper Range Value

**Description:**

This command is similar in effect to push the SPAN button of the HART device but doing it remotely through the HART network.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 37: Set Lower Range Value

**Description:**

This command is similar in effect to push the ZERO button of the HART device but doing it remotely through the HART network.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 38: Reset Configuration Changed Flag

**Description**:

This command resets the status bit that indicates that configuration has been changed.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 39: EEPROM Control

**Description:**

This command operates over the EEPROM changing its settings.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | EEPROM Control Code | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | EEPROM Control Code | 0 |

## COMMAND 40: Enter Exit Fixed Current Mode

**Description:**

This command writes the damping value of a HART device.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point Current (mA) | |
| 1 | | |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point Current (mA) | |
| 1 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 41: Perform Transmitter Self Test

**Description:**

This command starts the HART device Self Test to find if there is any problem with the device hardware.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 42: Perform Master Reset

**Description:**

This command performs a master reset in the HART device.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 43: Set PV Zero

**Description:**

This command forces the Primary Value to Zero.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 44: Write PV Units

**Description:**

This command changes the setting of the Primary Variable units.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Primary Variable Units Code | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Primary Variable Units Code | 0 |

## COMMAND 45: Trim DAC Zero

**Description:**

This command is used to calibrate the device Digital to analog converter zero. This is done by writing the same current value that is measured in the loop.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point measured current (mA) | |
| 1 | | |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point Current (mA) | |
| 1 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 46: Trim DAC Gain

**Description:**

This command is used to calibrate the device Digital to analog converter gain. This is done by writing the same current value that is measured in the loop.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point Measured Current (mA) | |
| 1 | | |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Floating Point Current (mA) | |
| 1 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |

## COMMAND 47: Write Transfer Function

**Description:**

This command changes the setting of the Transfer Function of the HART device.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transfer Function Code | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Transfer Function Code | 0 |

## COMMAND 48: Read Additional Transmitter Status

**Description:**

This commands gets extended information about the status of the Transmitter. This information is specific for each transmitter.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | Additional Status (Byte 0) | Additional Status (Byte 1) |
| **2** | Additional Status (Byte 2) | Additional Status (Byte 3) |
| . . . | . . . | |
| **13** | Additional Status (Byte 24) | 0 |

## COMMAND 49: Write PV Sensor Serial Number

**Description:**

This command changes the sensor serial number of the Primary Variable.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | Sensor Serial Number (Byte 0) | Sensor Serial Number (Byte 1) |
| **1** | Sensor Serial Number (Byte 2) | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | Sensor Serial Number (Byte 0) | Sensor Serial Number (Byte 1) |
| **2** | Sensor Serial Number (Byte 2) | 0 |

## COMMAND 50: Read Dynamic Variable Assignments

**Description:**

This command gets the actual assignment of the Dynamic Variables returned with command 3.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Transmitter Variable Code For Primary Variable | Transmitter Variable Code For Second Variable |
| 2 | Transmitter Variable Code For Third Variable | Transmitter Variable Code For Fourth Variable |

## COMMAND 51: Write Dynamic Variable Assignments

**Description:**

This command sets the assignment of the Dynamic Variables returned with command 3.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Code For Primary Variable | Transmitter Variable Code For Second Variable |
| 1 | Transmitter Variable Code For Third Variable | Transmitter Variable Code For Fourth Variable |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Transmitter Variable Code For Primary Variable | Transmitter Variable Code For Second Variable |
| 2 | Transmitter Variable Code For Third Variable | Transmitter Variable Code For Fourth Variable |

## COMMAND 52: Set Transmitter Variable Zero

**Description:**

This command forces a selected transmitter variable to zero.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Code | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Primary Variable Units Code | 0 |

## COMMAND 53: Write Transmitter Variable Units

**Description:**

This command changes a selected transmitter variable unit.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Code | Transmitter Variable Units Code |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Primary Variable Units Code | Transmitter Variable Units Code |

## COMMAND 54: Read Transmitter Variable Information

**Description:**

This commands gets information about any selected transmitter variable sensor.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Code | |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Upper Limit | |
| 1 | | |
| 2 | Transmitter Variable Lower Limit | |
| 3 | | |
| 4 | Transmitter Variable Damping Value (Sec) | |
| 5 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Transmitter Variable Code | Sensor Serial Number 0 |
| 1 | Sensor Serial Number 1 | Sensor Serial Number 2 |
| 2 | Unit Code for Limits | 0 |

## COMMAND 55: Write Transmitter Variable Damping Value

**Description:**

This command writes the Damping Value of a user selected transmitter variable.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Code | Transmitter Variable Damping Value (Byte 0) |
| 1 | Transmitter Variable Damping Value (Byte 1) | Transmitter Variable Damping Value (Byte 2) |
| 2 | Transmitter Variable Damping Value (Byte 3) | |

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Transmitter Variable Damping Value | |
| 1 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Transmitter Variable Code | 0 |

## COMMAND 56: Write Transmitter Variable Sensor Serial Number

**Description:**

This command writes the Serial Number of a user selected transmitter variable.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | Transmitter Variable Code | Transmitter Variable Sensor Serial Number (Byte 0) |
| **1** | Transmitter Variable Sensor Serial Number (Byte 1) | Transmitter Variable Sensor Serial Number (Byte 2) |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | Transmitter Variable Code | Transmitter Variable Sensor Serial Number (Byte 0) |
| **2** | Transmitter Variable Sensor Serial Number (Byte 1) | Transmitter Variable Sensor Serial Number (Byte 2) |

## COMMAND 57: Read Unit Tag Descriptor Date

**Description:**

This command reads an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | TAG ASCII Character 0 | TAG ASCII Character 1 |
| **2** | TAG ASCII Character 2 | TAG ASCII Character 3 |
| **3** | TAG ASCII Character 4 | TAG ASCII Character 5 |
| **4** | TAG ASCII Character 6 | TAG ASCII Character 7 |
| **5** | Descriptor ASCII Character 0 | Message ASCII Character 1 |
| . . . | . . . | |
| **12** | Descriptor ASCII Character 14 | Message ASCII Character 15 |
| **13** | Date | Date |
| **14** | Date | 0 |

## COMMAND 58: Write Unit Tag Descriptor Date

**Description:**

This command writes an ASCII Tag which identifies the device, an ASCII descriptor of the device and the last Date it has been configured.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | Packed ASCII TAG Byte 0 | Packed ASCII TAG Byte 1 |
| **1** | Packed ASCII TAG Byte 2 | Packed ASCII TAG Byte 3 |
| **2** | Packed ASCII TAG Byte 4 | Packed ASCII TAG Byte 5 |
| **3** | Packed ASCII Descriptor Byte 0 | Packed ASCII Descriptor Byte 1 |
| **.<br>.<br>.** | **.<br>.<br>.** | |
| **8** | Packed ASCII Descriptor Byte 10 | Packed ASCII Descriptor Byte 11 |
| **9** | Date | Date |
| **10** | Date | 0 |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| **0** | STATUS WORD | |
| **1** | TAG ASCII Character 0 | TAG ASCII Character 1 |
| **2** | TAG ASCII Character 2 | TAG ASCII Character 3 |
| **3** | TAG ASCII Character 4 | TAG ASCII Character 5 |
| **4** | TAG ASCII Character 6 | TAG ASCII Character 7 |
| **5** | Descriptor ASCII Character 0 | Message ASCII Character 1 |
| **.<br>.<br>.** | **.<br>.<br>.** | |
| **12** | Descriptor ASCII Character 14 | Message ASCII Character 15 |
| **13** | Date | Date |
| **14** | Date | 0 |

## COMMAND 59: Write Number Of Response Preambles

**Description:**

This command sets the number of preambles that the HART slave will use in every command response.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Number of Response Preambles | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Number of Response Preambles | 0 |

## COMMAND 108: Write Burst Mode Command Number

**Description:**

This command sets the command number that the HART device will use in Burst Mode.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Burst Mode Command Number | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Burst Mode Command Number | 0 |

## COMMAND 109: Burst Mode Control

**Description:**

This command sets the HART device Burst Mode.

**Write Parameters:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Burst Mode Control Code | |

**Floating Point Data Returned:**

NONE

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Burst Mode Control Code | 0 |

## COMMAND 110: Read All Dynamic Variables

**Description:**

This command gets all dynamic Variables.

**Write Parameters:**

NONE

**Floating Point Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | Primary Variable | |
| 1 | | |
| 2 | Second Variable | |
| 3 | | |
| 4 | Third Variable | |
| 5 | | |
| 6 | Fourth Variable | |
| 7 | | |

**Integer Data Returned:**

| WORD | HIGH BYTE | LOW BYTE |
|---|---|---|
| 0 | STATUS WORD | |
| 1 | Primary Variable Units Code | Second Variable Units Code |
| 2 | Third Variable Units Code | Fourth Variable Units Code |

# Appendix B – ProLinx HART Field Connections

HART field instruments are divided into four categories; A through D, depending on power requirements and whether the device sinks or sources a bias or operating current from the network. The following table defines the categories:

| Type | Description |
|---|---|
| A | Type A field instruments sink direct current from the network and receive operating power from the network. |
| B | Type B field instruments sink direct current from the network but receive no operating power from the network. |
| C | Type C field instruments source direct current to the network and receive no operating power from the network. |
| D | Type D field instruments neither source nor sink direct current from the network, nor receive operating power from the network. |

On a given network, A, B, and D devices may be combined. C and D devices may also be combined. A or B field instruments should not be combined with C devices.

## Network Power Supply Requirements

| Type | Network Power Supply |
|---|---|
| A | Required |
| B | Required |
| C | Must not be used. |
| D | Not required, but may be used. |

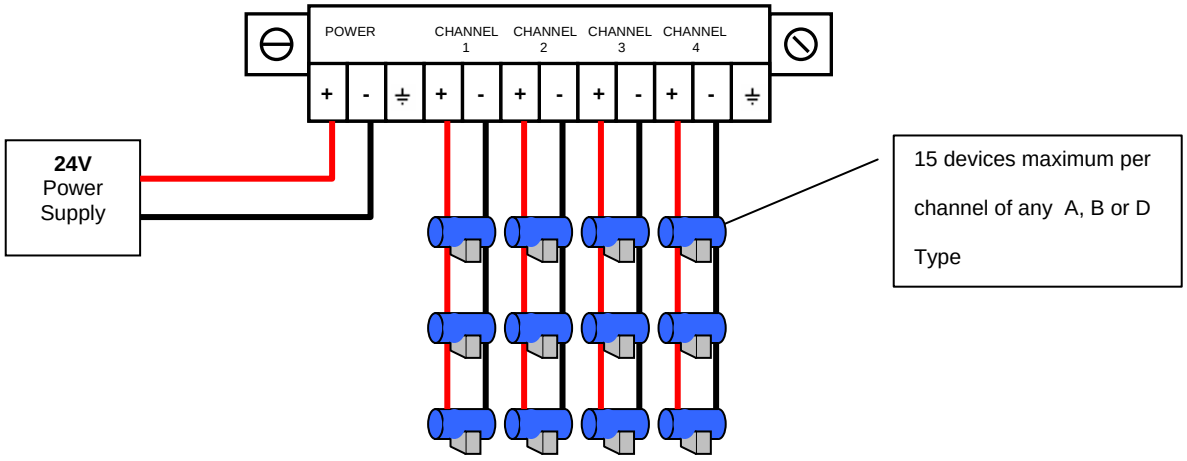
**Note:** The ProSoft 4107/4207 HART interface may be used with any HART field device.

## Connections for HART Devices (Types A, B, C and D)

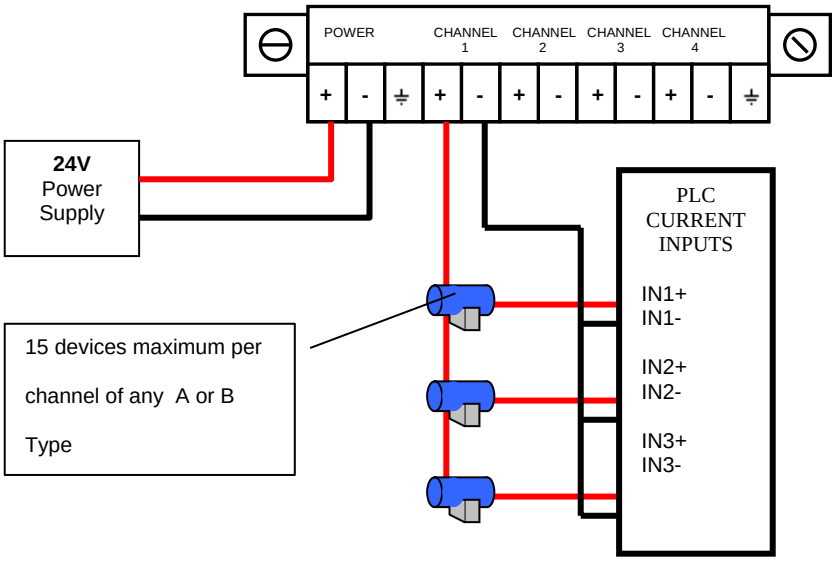
The ProLinx HART interface card has power supply input terminals for powering the devices in a network. This external power supply must be properly sized to supply current to all the HART devices in the network. A good rule of thumb is that each HART device may require as much as 20mA. Therefore, if you plan to place 10 HART devices in a network, you must provide a 24VDC power supply that can supply at least 200mA of output current.

**WARNING:** Whenever the HART module is used in multi-drop mode (2 to 15 instruments per channel), each of the instruments 4 to 20mA analog signal is lost.

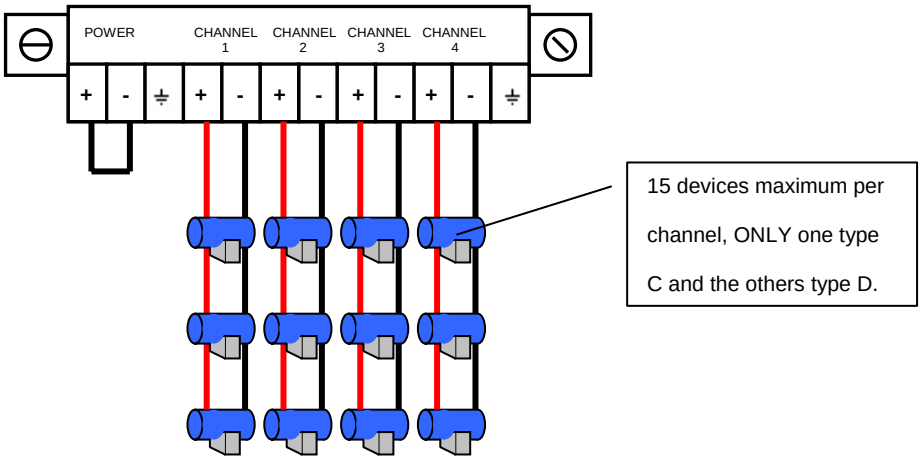
### ProLinx HART Connection Diagram for Multiple Loop Powered Devices, <u>Digital Signal Only</u> (A, B, or D types)



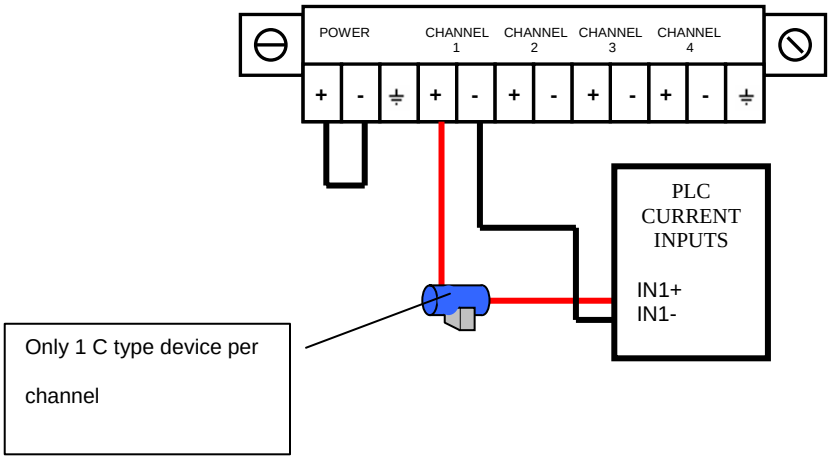
## ProLinx HART Connection Diagram for Multiple Loop Powered Devices, Digital and Analog Signal (A or B Types)



## ProLinx HART Connection Diagram for Multiple Self-Powered Devices, <u>Digital Signal Only</u> (C or D Types)



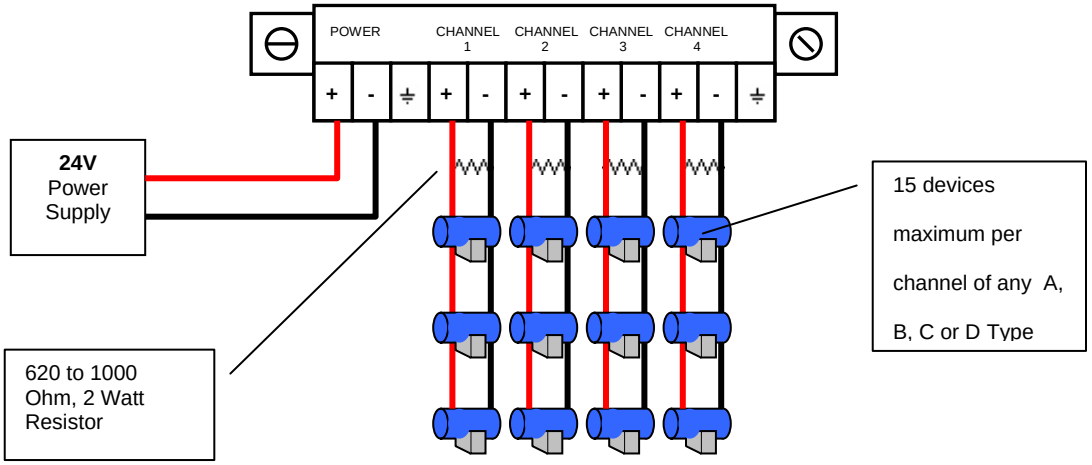
## ProLinx HART Connection Diagram for Single Self-Powered Device, Digital and Analog Signal (C Type only)



## ProLinx HART Connection Diagram for Multiple Single and Loop-Powered Devices, Digital Signal Only (A, B, C, or D Types)

This type of connection is NOT recommended, but has been used to mix A, B, C, and D types of HART devices.

**Note:** This type of connection may have problems associated with the lowering of AC impedance, lower immunity to electrical noise, and lower supply voltage to A and B type devices.


POWER
CHANNEL 1
CHANNEL 2
CHANNEL 3
CHANNEL 4
+ - ⏚ + - + - + - + - ⏚
24V Power Supply
620 to 1000 Ohm, 2 Watt Resistor
15 devices maximum per channel of any A, B, C or D Type